

# Tooltip

A flexible and accessible tooltip component for providing contextual information, guidance, or feedback. Supports multiple positions, arrow alignments, semantic variants, smooth animations, and advanced content, ensuring both usability and accessibility.

## How to use

### Basic Usage

A simple tooltip attached to any element.

```
import { AavaTooltipDirective } from "@aava/play-core";
```

### Positions

Tooltips can be positioned relative to the target element: top (default), bottom , left , right .

Arrow alignment options: start , center (default), end .

```
<aava-button
  label="Tooltip with Hover"
  variant="primary"
  size="small"
  state="default"
  aavaTooltipDescription="Tooltip with Hover"
  aavaTooltipType="simple"
  aavaTooltipTrigger="hover"
  aavaTooltipPosition="top"
  aavaTooltipArrow="center"
>
</aava-button>
```

### Variants

```

<button
  label=" Start"
  variant="primary"
  size="small"
  state="default"
  tooltipDescription=" Start"
  tooltipType="simple"
  tooltipTrigger="hover"
  tooltipPosition="top"
  tooltipArrow="start"
</button>
<button
  label=" Center"
  variant="primary"
  size="small"
  state="default"
  tooltipDescription=" Center"
  tooltipType="simple"
  tooltipTrigger="hover"
  tooltipPosition="top"
  tooltipArrow="center"
</button>
<button
  label=" End"
  variant="primary"
  size="small"
  state="default"
  tooltipDescription=" End"
  tooltipType="simple"
  tooltipTrigger="hover"
  tooltipPosition="top"
  tooltipArrow="end"
</button>

```

## API Reference

### Inputs (via config object)

Property	Type	Default	Description
content	string	"	Tooltip text or HTML content
position	'top'   'bottom'   'left'   'right'	'top'	Tooltip position relative to target
arrow	'start'   'center'   'end'	'center'	Arrow alignment

Property	Type	Default	Description
width	number	-	Target element width (for positioning, set automatically)
height	number	-	Target element height (for positioning, set automatically)
top	number	-	Target element top offset (set automatically)
left	number	-	Target element left offset (set automatically)
bottom	number	-	Target element bottom offset (set automatically)
enableAnimation	boolean	true	Enable/disable tooltip animation
behavior	'hover'   'focus'   'manual'	'hover'	How the tooltip is triggered

## Outputs

Tooltip is a presentational component. Events are handled by the directive or parent logic.

## Methods

Method	Parameters	Description
updatePosition()	-	Recalculates and updates tooltip position

# CSS Custom Properties

Tooltip uses CSS custom properties for theming and design tokens:

Property	Description
--tooltip-background	Tooltip background color
--tooltip-text	Tooltip text color
--tooltip-font	Font family for tooltip
--tooltip-padding	Padding inside tooltip
--tooltip-border-radius	Border radius for tooltip
--tooltip-shadow	Box shadow for tooltip
--tooltip-border-color	Border color for tooltip
--tooltip-z-index	Z-index for tooltip
--tooltip-info-background	Info variant background
--tooltip-info-text	Info variant text color
--tooltip-info-arrow	Info variant arrow color
--tooltip-success-background	Success variant background
--tooltip-success-text	Success variant text color
--tooltip-success-arrow	Success variant arrow color
--tooltip-warning-background	Warning variant background
--tooltip-warning-text	Warning variant text color
--tooltip-warning-arrow	Warning variant arrow color
--tooltip-error-background	Error variant background
--tooltip-error-text	Error variant text color
--tooltip-error-arrow	Error variant arrow color

## Accessibility

Tooltip follows accessibility best practices:

- Keyboard accessible (triggered by focus/hover)
- Proper ARIA attributes for screen readers
- Focus management and visible indicators
- High contrast and reduced motion support

## Theming & Design Tokens

All colors, spacing, and effects are controlled via semantic design tokens and CSS custom properties. Override these in your theme or component styles for custom branding.

## Best Practices

- Use tooltips for short, contextual information only
- Avoid placing critical information solely in tooltips
- Ensure tooltips are accessible via keyboard and screen readers
- Use semantic variants to match the context (info, warning, etc.)
- Test tooltip placement in responsive layouts
- Avoid excessive custom styling; use built-in variants and sizes